

DESCRIPTION

INFORMATION PROCESSING SERVER, REMOTE CONTROL
SYSTEM, AND REMOTE CONTROL METHOD

5

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is based upon and claims the benefit of
priority from the prior Japanese Patent Applications No.
2005-167231 filed on June 7, 2005, the entire contents of
10 which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to a technique for causing a
15 plurality of computers to conduct cooperative operation.

Related Art

In recent years, remote control has been made possible.
In the remote control, screen information supplied from a
20 remote computer is displayed on a terminal at hand, and the
remote computer is controlled using a mouse or a keyboard
connected to the terminal at hand. As examples thereof, "X
window system" developed by X.org Foundation, a function in
Microsoft Corporation called "Remote Desktop," and VNC (Virtual
25 Network Computing) software can be mentioned. In the X
window system, a view on the remote computer is drawn on the
terminal at hand by transmitting and receiving drawing
commands. However, it is also possible to construct a similar
system by encoding the screen on the remote computer by
30 using MPEG2-TS and decoding and displaying a result obtained
by the encoding, on the terminal at hand.

Even if the distance between two points is long, fast
communication has been made possible by advancement of the
Internet.

35 In general, a computer including a fast CPU and a large
capacity hard disk is heavy and it is not handy to carry. On the

other hand, a lightweight computer suitable for carrying is relatively slow in CPU capability and small in capacity of the hard disk.

Using the remote control scheme on the Internet,
5 however, it is possible to control the remote computer as if it is at hand. While the lightweight terminal at hand is carried, it is possible to access data stored in the large capacity hard disk and conduct processing using the fast CPU.

According to a method described in Japanese Patent
10 Application Laid-Open Publication No. 2003-288536, map data is transferred from a remote map data server to a car navigation terminal device at hand and displayed on the car navigation terminal device. A destination place is input to the car navigation terminal at hand, and this information is sent to the
15 map data server. Map data of the desired place is received by the car navigation terminal. As a result, a map of the desired place can be displayed on the car navigation terminal at hand. It is possible to get along without equipping the car navigation terminal with a large capacity storage device.

20 In this way, almost all of computer processing and data storage is entrusted to the remote computer, and the terminal at hand of the user conducts only light processing as an IO device which presents a result of processing sent from the remote computer, to the user, or accepts a keyboard or mouse
25 input given by the user. The application area of such architecture is spreading as the network speed becomes faster in recent years. In this architecture, most processing is conducted by the remote computer. This results in an advantage that the terminal at hand can be constructed using
30 hardware having a low processing capability. In an example of this architecture, a system department of an enterprise manages the remote computer and members are treated as users to construct a computer system. In another example of the architecture, an ASP (application service provider) manages
35 the remote computer and provides service of lending computer resources to users. Even if application is sophisticated and a

higher calculation processing capability is needed in such a case, it suffices to increase only the processing capability of the remote computer and it is unnecessary to increase the processing capability of the terminal at hand of the user. This
5 results in an advantage that the user can be saved the labor of updating the hardware.

For example, the broadcast or multicast is used to find a nearby device and nearby service, like NetBIOS used in Windows (trade mark), which is an OS of Microsoft Corporation,
10 and UPnP standardized by UPnP Forum. The conventional remote control technique has a problem: if a remote computer does not belong to the same IP subnetwork as the terminal at hand does, broadcast packets cannot be transmitted and received between the remote computer and a computer in the
15 neighborhood of the terminal at hand, and the remote computer cannot find a device and service in the neighborhood of the terminal at hand. Even in the case of a protocol using the multicast for finding a device or service, the TTL of a multicast packet is set to a small value or the network does not support
20 transfer of a multicast packet in many cases. This results in a problem that the remote computer cannot find a device or service in the neighborhood of the terminal at hand. Even if there is an UPnP AV server that retains video contents in the neighborhood of the terminal at hand, therefore, the remote
25 computer cannot find the UPnP AV server, resulting in a problem that the video contents in the UPnP AV server cannot be displayed on the terminal at hand.

A similar problem exists in the above-described Japanese Patent Application Laid-Open Publication No. 2003-288536 as
30 well. The remote map data server cannot find a device in the neighborhood of the car navigation terminal. Therefore, information utilizing service provided by the device cannot be displayed on the car navigation terminal at hand.

It is possible in the conventional technique to set, for
35 example, an L2TP tunnel between the terminal at hand and the remote computer and conduct simulation so as to cause the

remote computer to belong to the same subnetwork as the terminal at hand does in order to find service in the neighborhood of the terminal at hand. In this case, however, data communication conducted when the remote computer uses
5 service in the neighborhood of the terminal at hand also passes through an L2TP tunnel. This results in a problem that the terminal at hand must conduct tunnel transfer processing of data communication with the remote computer. Especially, as the network becomes fast and the data quantity in data
10 communication becomes large, the processing load becomes heavy and the CPU power with which the terminal at hand should be equipped also increases, resulting in a problem.

If the quantity of data communication conducted by the remote computer is increased by sophistication of application,
15 the user must upgrade the processing capability of the terminal at hand. This results in a problem of complication.

SUMMARY OF THE INVENTION

According to an aspect of the present invention, there is
20 provided with an information processing server that communicates with a terminal device connected to a first network and is connected to a second network different from the first network, comprising: a tunnel setter configured to set a tunnel with the terminal device; a reception unit configured to
25 receive a broadcast or multicast packet output from one or more service providing servers on the first network, via the tunnel; a server finding unit configured to find service providing servers on the first network and services provided by the service providing servers, based on the received broadcasts or
30 multicast packets; a notification unit configured to notify the terminal device of the services found, via the tunnel or the second network; and a data communication unit configured to be responsive to an execution request of the service from the terminal device via the tunnel or the second network to conduct
35 data communication concerning the service with a service providing server providing the service via the second network.

According to an aspect of the present invention, there is provided with a remote control system including a terminal device connected to a first network and an information processing server connected to a second network, wherein the terminal device comprises: a first tunnel setter configured to set a tunnel with the information processing server; and a transfer unit configured to receive a broadcast or multicast packet output from one or more service providing servers on the first network and transmits the received broadcast or multicast packets to the information processing server via the tunnel, and the information processing server comprises: a second tunnel setter configured to set the tunnel with the terminal device; a reception unit configured to receive the broadcast or multicast packets from the terminal device via the tunnel; a server finding unit configured to find service providing servers on the first network and services provided by the service providing servers, based on the received broadcast or multicast packets; a notification unit configured to notify the terminal device of the services found, via the tunnel or the second network; and a data communication unit configured to be responsive to an execution request of the service from the terminal device via the tunnel or the second network to conduct data communication concerning the service with a service providing server providing the service via the second network.

According to an aspect of the present invention, there is provided with a remote control method using a terminal device connected to a first network and an information processing server connected to a second network, comprising: setting a tunnel between the terminal device and the information processing server; transmitting a broadcast or multicast packet output from one or more service providing servers on the first network to the information processing server via the tunnel to cause the information processing server to find the service providing servers and services provided by the service providing servers; notifying the terminal device of the services found, from the information processing server via the tunnel or the

second network; and if execution request of the service is received by the information processing server from the terminal device via the tunnel or the second network, conducting data communication concerning the service between a service providing server providing the service and the information processing server, via the second network.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a diagram showing a typical configuration of an embodiment according to the present invention;

FIG. 2 is a diagram showing operation of connection between a terminal device T and a main device B;

FIG. 3 shows a state in which a tunnel TUN is set between the main device B and the terminal device T;

FIG. 4 shows a format example of a session connection request;

FIG. 5 shows a format example of communication parameters;

FIG. 6 shows a format example of an authentication request;

FIG. 7 shows a format example of an authentication response;

FIG. 8 shows a format example of a session connection rejection;

FIG. 9 shows a format example of a session connection response;

FIG. 10 shows a format example of a session connection acknowledgement;

FIG. 11 is a diagram showing an operation sequence in the case where the UPnP is used;

FIG. 12 shows a configuration example of the terminal device T;

FIG. 13 shows a format example of an address notice;

FIG. 14 shows a format example of an encapsulated packet;

FIG. 15 shows a configuration example of the main

device B;

FIG. 16 shows another configuration example of the main device B;

FIG. 17 is a diagram showing the case where the terminal device T has moved from an access point AP1 to an access point AP2;

FIG. 18 shows a processing procedure for finding contents service by using the UPnP;

FIG. 19 shows a flow of processing conducted when the main device B conducts moving picture reproduction by utilizing contents service;

FIG. 20 shows an example of a display image of icons in a contents server; and

FIG. 21 shows an example of a display image of a contents list.

DETAILED DESCRIPTION OF THE INVENTION

Hereafter, embodiments of the present invention will be described with reference to the drawings.

FIG. 1 is a diagram showing a typical configuration of an embodiment according to the present invention.

An access point AP, a terminal device T, a printer (service providing server) P, a computer (service providing server) C, and a DHCP server D are connected to each other via wireless links (first network). The access point AP and a main device (information processing server) B are connected to each other via a network (second network) N.

The access point AP conducts wireless communication by typically using a wireless protocol prescribed in IEEE 802.11a, b or g. Alternatively, a different wireless scheme may also be used. Furthermore, the terminal device T, the printer P, the computer C, and the main device B conduct communication to each other by typically using IPv4 (Internet Protocol version 4, hereafter referred to as IP). Alternatively, the protocol may be, for example, IPv6 or another protocol. The network N typically includes at least one router. The access point AP has a function

of transferring packets from the network N to the wireless links and transferring packets from the wireless links to the network N. This transfer is conducted by the Ethernet bridging function without referring to header information in IPv4 packets.

5 Operation of connection between the terminal device T and the main device B will now be described with reference to FIG. 2.

After having established a wireless link with the access point AP, the terminal device T acquires an IP address from, for example, the DHCP server D by using a DHCP function. Upon
10 acquiring an IP address At, the terminal device T establishes a session with the main device B. A sequence used when establishing a session is shown in FIG. 2. The terminal device T transmits a session connection request toward the IP address
15 of the main device B previously retained (S1).

This session connection request contains an IP address (At) of the terminal device T, a user identifier of a user using the terminal device T, and communication parameters of the terminal device T. FIG. 4 shows a format example of the
20 session connection request.

Here, the communication parameters contain a protocol identifier, an encoding rate, and a screen size. These three items form an entry. A plurality of entries can be described. The terminal device T enumerates usable entries.

25 The number of described entries is described in a field indicating the number of entries.

FIG. 5 shows a format example of the communication parameters. In an example of the communication parameters, the protocol identifier, the encoding rate, and the screen size
30 are MPEG2-TS, 4 Mbps, and 680 × 480, respectively.

Upon receiving the session connection request, the main device B transmits an authentication request to the IP address of the terminal device T contained in the session connection request (S2). The authentication request contains an IP
35 address (Ab) of the main device B, an authentication algorithm identifier, and a challenge parameter. FIG. 6 shows a format

example of the authentication request.

Upon receiving the authentication request, the terminal device T calculates an authentication response value by using an algorithm indicated by the authentication algorithm identifier on the basis of the challenge parameter and a previously given password. After the calculation, the terminal device T transmits an authentication response toward the main device B (S3). For calculating the authentication response value, for example, a method of concatenating the challenge parameter and the password and finding a message digest by using an MD5 algorithm on the basis of the concatenated challenge parameter and password can be used. The MD5 algorithm is described in detail in IETF RFC1321.

This authentication response contains the IP address (At) of the terminal device T, an authentication algorithm identifier, the user identifier of the user using the terminal device T, a challenge parameter, and an authentication response value. FIG. 7 shows a format example of the authentication response.

Upon receiving the authentication response, the main device B calculates the authentication response value according to a procedure similar to that of the terminal device T, and ascertains whether the calculated authentication response value coincides with the authentication response value in the authentication response. In the case of noncoincidence, the main device B transmits a session connection rejection to the terminal device T. A format of the session connection rejection is shown in FIG. 8. In the case of coincidence, the main device B transmits a session connection response to the terminal device T (S4). This session connection response contains the IP address (Ab) of the main device B, a session port number of the main device, and communication parameters. In the communication parameters, parameters that are contained in the communication parameters described in the session connection request and that can be used by the main device B are enumerated. FIG. 9 shows a format example of the session connection response.

Upon receiving the session connection response, the terminal device T transmits a session connection acknowledgement to the main device B (S5).

This session connection acknowledgement contains the IP address (At) of the terminal device T, a session port number of the terminal device T, and communication parameters. In the communication parameters, the terminal device describes one entry that is contained in the communication parameters described in the session connection response and that is the most favorite. FIG. 10 shows a format example of the session connection acknowledgement.

Upon receiving the session connection acknowledgement, the main device B transmits operation screen data of the main device B toward the IP address and the session port number of the terminal device T in the session connection acknowledgement by using the UDP and using, for example, MPEG2-TS (S8). Upon receiving the operation screen data from the main device B, the terminal device T displays image data contained in the operation screen data on a display included in the terminal device T and outputs voice data contained in the operation screen data to a speaker included in the terminal device T.

Upon receiving data from a keyboard or mouse included in the terminal device T, the terminal device T transmits the input event information toward the IP address and the session port number of the main device B contained in the session connection response by using the UDP (S8).

Upon receiving the input event information from the terminal device T, the main device B conducts processing according to the input event information as if the input event information is input from the keyboard or mouse included in the main device B.

Owing to the processing described heretofore, it becomes possible to control the main device B by using an input device such as a keyboard or mouse included in the terminal device T. It becomes possible to display the screen of the main device B

on the terminal device T.

In addition, the main device B sets a tunnel between the main device B and the terminal device T by using the IP address of the terminal device T contained in the session connection
5 acknowledgement (S6). FIG. 3 shows a state in which a tunnel TUN has been set between the main device B and the terminal device T. In general, forming a virtual communication path between two points on the network by carrying a packet according to a certain protocol as a data portion of a packet
10 according to a different protocol or the same protocol is referred to as tunneling. The virtual communication path is referred to as tunnel. For example, it indicates carrying a packet of a protocol with regard to the link layer such as an Ethernet frame with the packet being included in a packet of a protocol with
15 regard to the network layer such as an IP packet. Here, it is supposed that a tunnel has been set by using, for example, L2TP. A concrete procedure of the L2TP (Layer 2 Tunneling Protocol) is described in detail in "VPN/VLAN textbook" written by Haruki Koretomo and edited by Multimedia Communication Research
20 Society, or an IETF draft "draft-ietf-12tpext-12tp-base-15." After the tunnel TUN has been set, the main device B acquires an IP address Ab' from the DHCP server D via the tunnel TUN (through the terminal device T) by using the DHCP function. Hereafter, this address is referred to as main device tunnel
25 address. Upon acquiring the main device tunnel address, the main device B transmits an address notice containing a value of the address to the terminal device T (S7).

FIG. 13 shows a format example of the address notice.

The address notice has a main device IP address field and
30 a main device Ether address field. A main device tunnel address is described in the main device IP address field. An Ether address of the main device is described in the main device Ether address field.

After the terminal device T has received the address
35 notice, the terminal device T transfers a packet directed to the main device tunnel address and multicast and broadcast packets

to the main device B via the tunnel TUN.

As a result, the main device B can belong to the same subnetwork as the terminal device T does, via the tunnel TUN. Therefore, the main device B can receive a broadcast packet
5 and a multicast packet arriving at the terminal device T, and transmit a broadcast packet and a multicast packet to the same range as that of the terminal device T. Hereafter, this will be described concretely by taking UPnP as an example.

FIG. 11 is a diagram showing an operation sequence in
10 the case where the UPnP is used. In FIG. 11, portions using the tunnel TUN are indicated by dotted lines; whereas other portions are indicated by solid lines.

For example, it is now supposed that the computer C provides video contents by using the UPnP. The main device B
15 transmits SSDP SEARCH via the tunnel TUN by using the multicast to search for a computer that retains video contents (S11). The address Ab' is used as the source address of the SSDP SEARCH at this time. Upon receiving the SSDP SEARCH, the computer C returns an answer to the SSDP SEARCH toward
20 Ab' described in its source address (S12). As a result, the main device B can find the computer C.

The computer C periodically transmits SSDP
ADVERTISEMENT by using multicast to advertise itself. The
terminal device T transfers SSDP ADVERTISEMENT to the main
25 device B via the tunnel TUN (S13). As a result, the main device B can receive the SSDP ADVERTISEMENT. Therefore, the main device B can find the computer C.

In addition, the main device B transmits a contents list request toward the computer C by using unicast (S14). By
30 using Ab as the source address at this time, the main device B can receive a contents list response from the computer C without passing through the terminal device T (S15).

The main device B transmits a contents request for contents contained in a contents list toward the computer C by
35 using unicast (S16). By using Ab as the source address at this time, its contents response (response from the computer C) can

be received without passing through the terminal device T (S17).

It is possible to decode the contents thus acquired in the main device B, encode the decoded contents into a format for the terminal T, and transmit a result to the terminal device T via the tunnel TUN or not via the tunnel TUN (i.e. via second network).

Here, operation of the main device B and the terminal device T has been described taking the UPnP as an example of a protocol to be used. However, the application range of the present invention is not restricted to the UPnP. There are a large number of protocols for finding a device by using the multicast or broadcast and then conducting communication with the found device by using the unicast, and the present invention can be applied to them. As a different example, NetBIOS/SMB used in Windows, which is an OS of Microsoft Corporation, can also be mentioned.

If a packet is sent from the computer C toward the address Ab' of the main device B, the terminal device T sends this packet to the main device B via the tunnel TUN. In a response packet to the packet having Ab' as its destination address, the address Ab' is described as the source address by the main device B.

A configuration example of the terminal device T will now be described with reference to FIG. 12.

The terminal device T includes a user input-output processor 11, a session setter 12, a tunnel setter 13, a tunnel processor 14, a packet classification decider 15, and a wireless transmission and reception unit 16.

The wireless transceiver unit 16 conducts establishment of a wireless link, processing of conducting communication with the DHCP server and acquiring an IP address, and packet transmission and reception processing to be conducted via the wireless link.

Upon receiving a notice to the effect that an IP address has been acquired, from the wireless transmission and reception

unit 16, the session setter 12 generates a session connection request containing the IP address received from the wireless transmission and reception unit 16, a user identifier of the user previously set, and communication parameters previously set, and transmits the session connection request toward the IP address of the main device B previously set, via the wireless transmission and reception unit 16.

Upon receiving the authentication request via the wireless transmission and reception unit 16, the session setter 12 generates an authentication response. The authentication response contains an IP address received from the wireless transmission and reception unit 16, an authentication algorithm identifier contained in the authentication request, a user identifier of the user previously set, a challenge parameter contained in the authentication request, and an authentication response value calculated using an algorithm indicated by the authentication algorithm identifier on the basis of the challenge parameter contained in the authentication request and a previously set password. The session setter 12 transmits the authentication response toward the source address and the source port of the authentication request via the wireless transmission and reception unit 16.

Upon receiving a session connection response via the wireless transmission and reception unit 16, the session setter 12 generates a session connection acknowledgement. The session connection acknowledgement contains an IP address received from the wireless transmission and reception unit 16, a session port number previously set, and predetermined communication parameters having the highest priority among the communication parameters contained in the session connection response. The session setter 12 transmits the session connection acknowledgement toward the source address and the source port of the authentication request via the wireless transmission and reception unit 16.

Upon receiving a control packet for setting the L2TP tunnel via the wireless transmission and reception unit 16, the

tunnel setter 13 sets an L2TP tunnel in accordance with communication specifications of the L2TP.

Upon setting the tunnel, the tunnel setter 13 delivers parameters of the L2TP tunnel to the tunnel processor 14. The
5 parameters contain a session ID, an address of the opposite party, a port number of the opposite party (a port number for tunnel data), an own address, and an own port number (a port number for tunnel data).

Upon receiving an address notice via the wireless
10 transmission and reception unit 16, the packet classification decider 15 stores a main device IP address (main device tunnel address) and a main device Ether address.

Furthermore, the packet classification decider 15 passes the value of the main device Ether address to the wireless
15 transmission and reception unit 16. Thereafter, the wireless transmission and reception unit 16 delivers Ether packets having the main device Ether address as the destination, Ether packets having a broadcast address as the destination, and Ether packets having an arbitrary multicast address as the
20 destination, to the packet classification decider 15.

If the terminal device T uses 802.11a, b or g at this time, the terminal device T couples a connection-relation to the Ether
address of the main device with the access point AP in order to
cause the access point AP to transmit the Ether packet having
25 the main device Ether address as the destination to the terminal device T.

Upon receiving the Ether packet having the main device Ether address as the destination, the Ether packet having a broadcast address as the destination, and the Ether packet
30 having an arbitrary multicast address as the destination from the wireless transmission and reception unit 16, the packet classification decider 15 delivers them to the tunnel processor 14.

Here, the packet classification decider 15 may deliver
35 only the Ether packets having the main device Ether address as the destination, the Ether packets having a broadcast address

as the destination, and the Ether packets having a previously set multicast address as the destination to the tunnel processor 14.

5 Upon receiving a packet capsulated according to the L2TP from the wireless transmission and reception unit 16 (from the main device B), the tunnel processor 14 removes the Ether header, IP header, UDP header and L2TP header from the head of the packet, and transmits the remaining Ether header, IP header and data to a wireless link via the wireless transmission and reception unit 16.

10 Upon receiving a packet from the packet classification decider 15, the tunnel processor 14 adds an Ether header, an IP header, a UDP header, and an L2TP header to the packet, and transmits a resultant packet to a wireless link (toward the main device B) via the wireless transmission and reception unit 16. The format of the capsulated packet (format of packet for L2TP tunnel) is shown in FIG. 14. At this time, the Ether address of the own node is described in the source address of the Ether header, and an Ether address of a router (not illustrated) is described in the destination address. Furthermore, the address of the opposite party received from the tunnel setter 13 is described in the destination address of the IP header, and the own address received from the tunnel setter 13 is described in the source address. Furthermore, the port number of the opposite party received from the tunnel setter 13 is described in the destination port of the UDP header, and the own port number received from the tunnel setter 13 is described in the source port. The session ID received from the tunnel setter 13 is described in the session ID of the L2TP header.

20 Upon receiving operation screen data encoded according to, for example, MPEG2-TS from the main device B via the wireless transmission and reception unit 16, the user input-output processor 11 decodes the data into image data and voice data, outputs the image data to the display included in the terminal device T, and outputs the voice data to the speaker included in the terminal device T.

25

30

35

Upon receiving an input from the keyboard or mouse included in the terminal device T, the user input-output processor 11 encodes the input, and transmits the result to the main device B via the wireless transmission and reception unit
5 16 as input event information.

FIG. 18 shows a processing procedure for finding contents service using the UPnP (Universal Plug and Play) conducted by the main device B. Here, a computer that provides contents service is referred to as contents server.

10 Upon finishing the setting of the tunnel TUN, the main device B transmits an SSDP SEARCH packet via the tunnel TUN (S21). At this time, the destination address of the SSDP SEARCH packet is 239.255.255.250, and the source address is the main device tunnel address Ab'. Since the SSDP SEARCH
15 packet is to be passed through the tunnel, the SSDP SEARCH packet is capsulated to the format shown in FIG. 14, and transmitted to the network N.

Subsequently, it is determined whether a response packet to the SSDP SEARCH packet has been received via the tunnel
20 TUN (S22).

If the response packet to the SSDP SEARCH packet has been received (YES of S22), update processing of the contents server list is conducted (S23). The contents server list contains (device name, device IP address, lifetime, and contents
25 directory control URL).

As for update processing of the contents server list, it is determined whether an entry having a device IP address that coincides with a source address of the response packet to the SSDP SEARCH packet is already present in the contents server
30 list. If such an entry is present, the lifetime of the entry is rewritten to become time indicated in a CACHE-CONTROL field in the response packet to the SSDP SEARCH packet. On the other hand, if such an entry is not present, a new entry is added to the contents server list.

35 (Source address, and time indicated in the CACHE-CONTROL field) in the response packet to the SSDP

SEARCH packet are described in (device IP address and lifetime) of the added entry. Values of (friendly Name and control URL) in a series of Description documents obtained by accessing a URL indicated by a LOCATION field in the response packet to the
5 SSDP SEARCH packet are described in (device name and contents directory control URL) of the added entry.

Thereafter, or if the response packet to the SSDP SEARCH packet is not received (NO of S22), a timer event is set in a timer within the main device B ten minutes later (S24), and a
10 packet reception and timer event reception waiting state is brought about (S25).

If a packet is received, it is determined whether the packet is an SSDP ADVERTISEMENT packet received via the tunnel TUN (S26). If the packet is an SSDP ADVERTISEMENT
15 packet (YES of S26), contents server list update processing is conducted (S27) and the packet reception and timer event reception waiting state is brought about again (S25).

In the contents server list update processing, it is determined whether an entry of a device IP address coinciding with a source address of the SSDP ADVERTISEMENT packet is
20 already present in the contents server list. If such an entry is present, lifetime in the entry is rewritten to become time indicated in a CACHE-CONTROL field of the SSDP ADVERTISEMENT packet.

25 If such an entry is not present, a new entry is added to the contents server list. (Source address, and time indicated in the CACHE-CONTROL field) in the SSDP ADVERTISEMENT packet are described in (device IP address and lifetime) of the added entry. Values of (friendly Name and control URL) in a series of
30 Description documents obtained by accessing a URL indicated by a LOCATION field in the SSDP ADVERTISEMENT packet are described in (device name and contents directory control URL) of the added entry.

If the packet is not the SSDP ADVERTISEMENT packet
35 (NO of S26), it is determined whether a timer event is received (S28). If the timer event is not received (NO of S28), the

packet reception and timer event reception waiting state is brought about again (S25).

On the other hand, if the timer event is received (YES of S28), timeout processing of the contents server list is conducted
5 (S29).

In the timeout processing of the contents server list, 10 minute is subtracted from the lifetime of each entry in the contents server list. In addition, entries that are 0 or less in lifetime after the subtraction are deleted.

10 Thereafter, the SSDP SEARCH packet is transmitted via the tunnel TUN again (S21).

Owing to the operation heretofore described, it becomes possible for the main device B to find contents service in the neighborhood of the terminal device T via the tunnel TUN and
15 retain a list of contents servers that provide the contents service.

FIG. 19 shows a flow of processing at the time when the main device B conducts moving picture reproduction by utilizing the contents service.

20 The main device B transmits display images (identifiers) of icons of contents servers of all entries in the contents list to the terminal device T (S31).

FIG. 20 shows a display image example of icons of contents servers transmitted by the main device B. In FIG. 20,
25 there are four entries in the contents list in the main device B, and device names are "Sunflower," "Cherry blossom," "Bamboo" and "Japanese apricot."

If a mouse click event is input from the terminal device T to the main device B (YES of S32), the main device B acquires a
30 contents list from a subject contents server subjected to mouse click (S33).

Acquisition of the contents list is conducted by transmitting a contents list acquisition request to a contents directory control URL of the pertinent entry in the contents
35 server list. The contents list acquisition request is transmitted without being passed through the tunnel. Its destination

address is a device IP address of the contents server, and its source address is the IP address Ab of the main device B.

After the main device B has acquired the contents list, the main device B transmits the display image (identifiers) of the contents list to the terminal device T (S34).

FIG. 21 shows an example of a display image of the contents list. Titles of two moving pictures, a title of one still picture, a title of one piece of music, and contents classifications retained by the contents server are displayed.

If a mouse click event is input from the terminal device T (YES of S35), the main device B transmits a contents acquisition request for a contents title that is the subject of the mouse click, and acquires contents (S36). The contents contain still picture data, moving picture data, voice data, document data, or an arbitrary combination of them.

The contents acquisition request is transmitted without being passed through the tunnel. Its destination address is a device IP address, and its source address is the IP address Ab of the main device B.

If the contents are, for example, a moving picture, the main device B transmits a displayed image of the moving picture to the terminal device T (S37).

A configuration example of the main device B will now be described with reference to FIG. 15.

The main device B includes an operation screen generator 21, a central processor 22, an external input processor 23, a session setter 24, a tunnel setter 25, a tunnel processor 26, and a wired transmission and reception unit 27.

The wired transmission and reception unit 27 conducts transmission and reception processing of a packet.

Upon receiving a packet from the outside, the wired transmission and reception unit 27 conducts inspection on the packet.

If a protocol number in its IP header indicates the UDP and its UDP destination port indicates a previously given number for session setting, the wired transmission and

reception unit 27 delivers the packet to the session setter 24.

If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a session port number given by the session setter 24 (a port number for data
5 communication using an established session), the wired transmission and reception unit 27 delivers the packet to the external input processor 23.

If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a previously given
10 number for tunnel setting, the wired transmission and reception unit 27 delivers the packet to the tunnel setter 25.

If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a number given by the tunnel setter 25 (a port number for tunnel data), the wired
15 transmission and reception unit 27 delivers the packet to the tunnel processor 26.

The wired transmission and reception unit 27 delivers other packets to the central processor 22.

Upon receiving a session connection request via the wired
20 transmission and reception unit 27, the session setter 24 transmits an authentication request toward the terminal device T via the wired transmission and reception unit 27. The authentication request contains a previously set IP address of the own node, a previously set authentication algorithm
25 identifier, and a previously set challenge parameter.

Upon receiving an authentication response via the wired transmission and reception unit 27, the session setter 24 calculates an authentication response value by using an algorithm indicated by the authentication algorithm identifier on
30 the basis of the challenge parameter contained in the authentication response and a password. The session setter 24 determines whether the calculated authentication response value coincides with an authentication response value contained in the authentication response.

35 In the case of noncoincidence, the session setter 24 generates a session connection rejection having a description of

a user identifier contained in the authentication response, and transmits the session connection rejection to the terminal device T via the wired transmission and reception unit 27.

5 If the two authentication response values coincide with each other, the session setter 24 generates a session connection response and transmits the session connection response to the terminal device T via the wired transmission and reception unit 27. The session connection response contains a previously set
10 IP address of the own node, a previously set port number (session port number) of the own node, and communication parameters that are common to a plurality of previously set communication parameters and communication parameters contained in the session connection request.

Upon receiving a session connection acknowledgement
15 via the wired transmission and reception unit 27, the session setter 24 delivers communication parameters, an IP address of the terminal device T and a session port number contained in the session connection acknowledgement to the operation screen generator 21.

20 Furthermore, the session setter 24 delivers the previously set port number (session port number) of the own node to the wired transmission and reception unit 27.

Upon receiving a control packet for L2TP tunnel setting via the wired transmission and reception unit 27, or after the
25 tunnel setter 25 has received the session connection acknowledgement, the tunnel setter 25 sets the L2TP tunnel according to the communication specifications of the L2TP.

If the tunnel is set, the tunnel setter 25 delivers a session ID, an address of the opposite party, a port number of
30 the opposite party (a port number for tunnel data), an own address, and an own port number (a port number for tunnel data), which are parameters of the L2TP tunnel, to the tunnel processor 26.

Furthermore, the tunnel setter 25 delivers the own port
35 number to the wired transmission and reception unit 27.

Upon receiving a packet capsulated according to the L2TP

from the wired transmission and reception unit 27, the tunnel processor 26 removes the Ether header, IP header, UDP header, and L2TP header from the head of the packet, and delivers the remaining Ether header, IP header and data to the central processor 22.

Furthermore, upon receiving an Ether packet from the central processor 22, the tunnel processor 26 adds an Ether header, an IP header, a UDP header and an L2TP header to the Ether packet. The tunnel processor 26 transmits a resultant Ether packet to the network N via the wired transmission and reception unit 27. At this time, an Ether address of the own node is described in a source address of the Ether header, and an Ether address of a router (not illustrated) is described in a destination address of the Ether header. Furthermore, the address of the opposite party received from the tunnel setter 25 is described in a destination address of the IP header, and the own address received from the tunnel setter 25 is described in a source address of the IP header. The port number of the opposite party received from the tunnel setter 25 is described in a destination port of the UDP header, and the own port number received from the tunnel setter 25 is described in a source port of the UDP header. The session ID received from the tunnel setter 25 is described in a session ID of the L2TP header.

Upon receiving input event information from the terminal device T via the wired transmission and reception unit 27, the external input processor 23 decodes the input event information and transmits the result to the central processor 22.

Upon receiving screen information and voice information from the central processor 22, the operation screen generator 21 encodes them according to the communication parameters received from the session setter 24, and transmits a result toward the IP address and the session port number received from the session setter 24, via the wired transmission and reception unit 27.

The central processor 22 is operating a predetermined program that operates according to the input event information

given from the external input processor 23. It is desirable that this program is a basic program. The central processor 22 operates, manipulates and stops a previously retained application program in accordance with the input event
5 information given from the external input processor 23.

As a result, the basic program or an application program can be operated as if the mouse or keyboard of the terminal device T is equipped with the main device.

The central processor 22 outputs the state of the basic
10 program or the application program which changes according to the input event information, to the operation screen generator 21 as image information or voice information.

As a result, an image that would be displayed on a display if the display were equipped with the main device B can
15 be displayed on the terminal device T. Furthermore, a voice that would be output by a speaker if the speaker were equipped with the main device B can be output from the terminal device T.

The central processor 22 delivers the multicast packets
20 and broadcast packets generated on the operation process and response packets to unicast packets received via the tunnel processor 26 to the tunnel processor 26, and delivers other packets to the wired transmission and reception unit 27.

In a packet transmitted via the tunnel processor 26, a
25 source address of the inside IP header is Ab'. A destination address of the inside IP header is a communication destination address of the central processor 22. A source address of an outside IP header is Ab. A destination address of the outside IP header is At.

30 In a packet transmitted without being passed through the tunnel processor 26, a source address of the IP header is Ab, and a destination address of the IP header is a communication destination address of the central processor 22.

Furthermore, the central processor 22 causes the basic
35 program or application program to operate according to a packet received from the external input processor 23 or the

tunnel processor 26.

Another configuration example of the main device B will now be described with reference to FIG. 16.

The main device roughly includes a communication
5 processor 31 and a computing unit 32. The computing unit 32 is, for example, a personal computer. The communication processor 31 is, for example, a box attached to the outside of the computing unit 32. The communication processor 31 has a function of making it seem to the computing unit 32 that the
10 computing unit 32 belongs to the same subnetwork as the terminal T does.

The computing unit 32 includes a CPU, a memory, an OS, and application programs. The computing unit 32 further includes a communication interface, a keyboard interface, a
15 mouse interface, a display interface, and a speaker interface.

The communication interface, the keyboard interface, the mouse interface, the display interface, and the speaker interface are connected to the communication processor 31.

The communication interface acquires an IP address
20 according to the DHCP via the communication processor 31 and the terminal device T, and conducts communication with another device via the communication processor 31 by using this IP address.

The communication processor 31 includes a pseudo Ether
25 unit 33, a pseudo keyboard 34, a pseudo mouse 35, a pseudo display 36, a pseudo speaker 37, a tunnel setter 38, a tunnel processor 39, a NAT processor 40, a session processor 41, a session setter 42, and a wired transmission and reception unit 43.

30 The wired transmission and reception unit 43 conducts packet transmission and reception processing.

Upon receiving a packet from the outside, the wired transmission and reception unit 43 conducts inspection on the packet.

35 If a protocol number in its IP header indicates the UDP and its UDP destination port indicates a previously given

number for session setting, the wired transmission and reception unit 43 delivers the packet to the session setter 42.

5 If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a session port number given by the session setter 42, the wired transmission and reception unit 43 delivers the packet to the session processor 41.

10 If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a previously determined number for tunnel setting, the wired transmission and reception unit 43 delivers the packet to the tunnel setter 38.

15 If the protocol number in its IP header indicates the UDP and its UDP destination port indicates a number given by the tunnel setter 38, the wired transmission and reception unit 43 delivers the packet to the tunnel processor 39.

The wired transmission and reception unit 43 delivers other packets to the NAT processor 40.

20 Upon receiving a session connection request via the wired transmission and reception unit 43, the session setter 42 transmits an authentication request toward the terminal device T via the wired transmission and reception unit 43. The authentication request contains a previously set IP address of the own node, a previously set authentication algorithm identifier, and a previously set challenge parameter.

25 Upon receiving an authentication response via the wired transmission and reception unit 43, the session setter 42 calculates an authentication response value by using an algorithm indicated by the authentication algorithm identifier on the basis of the challenge parameter contained in the authentication response and a previously set password. The session setter 42 determines whether the calculated authentication response value coincides with an authentication response value contained in the authentication response.

35 In the case of noncoincidence, the session setter 42 generates a session connection rejection (FIG. 8) having a description of a user identifier contained in the authentication

response, and transmits the session connection rejection to the terminal device T via the wired transmission and reception unit 43.

5 If the two authentication response values coincide with each other, the session setter 42 generates a session connection response and transmits the session connection response to the terminal device T via the wired transmission and reception unit 43. The session connection response contains a previously set IP address of the own node, a previously set port number
10 (session port number) of the own node, and communication parameters that are common to a plurality of previously set communication parameters and communication parameters contained in the session connection request.

15 Upon receiving a session connection acknowledgement via the wired transmission and reception unit 43, the session setter 42 delivers communication parameters, an IP address of the terminal device T and a session port number contained in the session connection acknowledgement to the session processor 41.

20 Furthermore, the session setter 42 delivers the previously set port number (session port number) of the own node to the wired transmission and reception unit 43.

25 Upon receiving input event information from the terminal device T via the wired transmission and reception unit 43, the session processor 41 decodes the input event information, delivers input information supplied from the mouse of the terminal device T to the pseudo mouse processor 35, and delivers input information supplied from the keyboard of the terminal device T to the pseudo keyboard processor 34.

30 Upon receiving image information from the pseudo display processor 36 or voice information from the pseudo speaker processor 37, the session processor 41 encodes the image information or the voice information in accordance with the communication parameters received from the session setter
35 42, and transmits the encoded image information or the voice information toward the IP address and port number received

from the session setter 42 via the wired transmission and reception unit 43.

Upon receiving a keyboard input from the session processor 41, the pseudo keyboard 34 converts the keyboard
5 input to a signal of the keyboard interface in the computing unit 32, and delivers the signal to the computing unit 32.

Upon receiving a mouse input from the session processor 41, the pseudo mouse 35 converts the mouse input to a signal of the mouse interface in the computing unit 32, and delivers
10 the signal to the computing unit 32.

If a image signal is input from the display interface in the computing unit 32, the pseudo display 36 converts the image signal to an internal form, and outputs a resultant signal to the session processor 41 as image information.

15 If a voice signal is input from the speaker interface in the computing unit 32, the pseudo speaker 37 converts the voice signal to an internal form, and outputs a resultant signal to the session processor 41 as voice information.

Since the tunnel setter 38 is the same as the tunnel
20 setter 25 described with reference to FIG. 15, detailed description of the tunnel setter 38 will be omitted.

In brief, the pseudo Ether unit 33 has a function of returning a response packet to a packet brought through the tunnel processor 39, via the tunnel processor 39, and returning
25 a response packet to a packet brought through the NAT processor 40, via the NAT processor 40. Hereafter, details of the function will be described.

The pseudo Ether unit 33 converts a packet received from the tunnel processor 39 or the NAT processor 40 to a signal of the Ether interface in the computing unit 32, and delivers the
30 resultant signal to the computing unit 32.

Upon receiving a packet signal from the Ether interface in the computing unit 32, the pseudo Ether unit 33 converts the packet signal to an internal form. If a flow to which the packet
35 is belonging is stored, the pseudo Ether unit 33 delivers the resultant signal to the tunnel processor 39. Otherwise, the

pseudo Ether unit 33 delivers the resultant signal to the NAT processor 40.

If a packet is input from the tunnel processor 39 and its destination address is a unicast address, the pseudo Ether unit
5 33 retains an entry having (destination address, source address, protocol number, destination port number, and source port number). Thereby, storage of a flow is conducted. Here, the destination address, source address and protocol number are described in fields in the IP header. The destination port
10 number and source port number are described in the UDP or TCP header. The pseudo Ether unit 33 can store as many different entries as a predetermined number.

Upon receiving a packet signal from the computing unit 32, the pseudo Ether unit 33 determines whether (source
15 address, destination address, protocol number, source port number, and destination port number) of the packet coincide with (destination address, source address, protocol number, destination port number, and source port number) of the stored flow. In the case of coincidence, the pseudo Ether unit 33
20 judges the packet to belong to the flow.

If the pseudo Ether unit 33 has not received a packet belonging to the flow from the computing unit 32 for a predetermined time and the pseudo Ether unit 33 has not
received a packet having the same (destination address, source
25 address, protocol number, destination port number, and source port number) as the flow does, from the tunnel processor 39 for a predetermined time, the stored entry is erased.

Here, (destination address, source address, protocol number, destination port number, and source port number) have
30 been used as an example of the flow discrimination. If the protocol number does not indicate the UDP or TCP, however, processing can be conducted by setting the destination port number to 0 and the source port number to 0. In the case of IPv6, it is also possible to discriminate the flow by using a flow
35 label in its IP header.

Upon receiving a packet encapsulated according to the L2TP

from the wired transmission and reception unit 43, the tunnel processor 39 removes the Ether header, IP header, UDP header, and L2TP header from the head of the packet, and delivers the remaining Ether header, IP header and data to the pseudo Ether unit 33.

Furthermore, upon receiving an Ether packet from the pseudo Ether unit 33, the tunnel processor 39 adds an Ether header, an IP header, a UDP header and an L2TP header to the Ether packet. The tunnel processor 39 transmits a resultant Ether packet to the wired transmission and reception unit 43. At this time, an Ether address of the own node is described in a source address of the Ether header, and an Ether address of a router (not illustrated) is described in a destination address of the Ether header.

Furthermore, the tunnel processor 39 describes the address of the opposite party received from the tunnel setter 38 in a destination address of the IP header, and described the own address received from the tunnel setter 38 in a source address of the IP header.

Furthermore, the tunnel processor 39 describes the port number of the opposite party received from the tunnel setter 38 in a destination port of the UDP header, and describes the own port number received from the tunnel setter 38 in a source port of the UDP header.

Furthermore, the tunnel processor 39 describes the session ID received from the tunnel setter 38 in a session ID of the L2TP header.

Upon receiving a packet from the pseudo Ether unit 33, the NAT processor 40 replaces a source address of the Ether header with an Ether address of the own node, replaces a source address of the IP header with an IP address of the own node, and transmits the resultant packet to the wired transmission and reception unit 43.

Upon receiving a packet from the wired processor 43, the NAT processor 40 replaces the destination address of the Ether header with an Ether address of the computing unit 32, replaces

a destination address of the IP header with an IP address of the computing unit 32, and delivers the resultant packet to the pseudo Ether unit 33.

As heretofore described, the main device B includes the
5 communication processor 31 and the computing unit 32.
Therefore, the computer function 32 may not have a function of
finding service in the neighborhood of the terminal device T. In
other words, the communication processor 31 delivers a packet
10 for finding service in the neighborhood of the terminal device T
to the computing unit 32 by setting a tunnel. As a result, the
computing unit 32 can find service in the neighborhood of the
terminal device T.

Furthermore, the computing unit 32 can use service
found via the communication processor 31.

15 Furthermore, the computing unit 32 can transmit a
screen display image and a speaker output to the terminal
device T and receive mouse operation and keyboard operation
from the terminal device T via the communication processor 31.

As a result, an ordinary personal computer can be used
20 as the computing unit 32. The price of the main device B can
be lowered by thus using a general-purpose computer. Even if
the data processing capability of the main device B becomes
insufficient, it is sufficient to replace only the computing unit 32.
Therefore, the cost for updating the data processing capability
25 can be reduced.

Heretofore, the case where the main device B physically
includes the two units, i.e., the computing unit 32 and the
communication processor 31 has been described. Instead, the
present invention can also be applied to the case where the
30 main device B is one apparatus in hardware and the computing
unit 32 and the communication processor 31 are separated from
each other in software. For example, it is also possible to
implement the communication processor 31 by using a host OS
that directly controls hardware while using software that
35 implements a virtual PC such as VMWARE, and regard a guest
OS that operates on the virtual PC implemented on the host OS

as the computing unit 32.

In this case, it becomes possible to implement a plurality of computing units 32 by using one piece of hardware.

Here, if the terminal device T has moved from an access point AP1 to an access point AP2 as shown in FIG. 17, the terminal device T transmits a DHCP RENEW packet having an address acquired from a DHCP server D1, at the time of re-connection of a wireless link with AP 2 after the movement. An answer to this is not obtained for a predetermined time. As a result, the terminal device T starts the operation for re-acquiring the address.

By the operation, the terminal device T can acquire an address from a DHCP server D2.

After the address re-acquisition, the terminal device T establishes a session and a tunnel between the main device B and the terminal device T.

In the embodiments of the present invention heretofore described, the terminal device T conducts packet transmission and reception via a wireless link. Even if the terminal device T conducts packet transmission and reception via a wired link, the effects of the present invention are not diminished.

The L2TP is used as an example of the tunnel. Even if a different tunnel technique, such as PPTP(Point-to-Point Tunneling Protocol) or MPLS(Multi-Protocol Label Switching), is used instead of the L2TP, however, the effects of the present invention are not diminished.

The DHCP is used as an example of means by which the main device B or the terminal device T acquires the address Ab' or At. Even if other means such as a method of previously conducting manual setting are used instead of the DHCP, the effects of the present invention are not diminished.

In the case where the IPv6 is used, it is also possible to use the automatic address setting function instead of the DHCP. In this case, the main device B transmits an RS (Router Solicitation) message to the network having the terminal device, via the tunnel, and receives an RA (Router Advertisement)

message from a router (not illustrated) via the tunnel. As a result, the main device B acquires an IPv6 address. Or in the case IPv6, the main device B can also use a link local address.

5 An example in which the main device B or the terminal device T uses a different address has been described. Alternatively, the same address can also be used. In this case, the address of the terminal device T is contained in the session connection acknowledgement, and a notice of a resultant session connection acknowledgement is sent to the main device
10 B. The terminal device T transmits packets other than operation screen data supplied from the main device B, packets received from the tunnel, and a response packet to an ARP request transmitted by itself, to the main device B via the tunnel.

15 If power supplied to the terminal device is turned off, the main device can detect it by using, for example, the control function of the L2TP.

Even if the power turn-off of the terminal device is detected when the main device is conducting communication with, for example, the computer C without using the tunnel, the
20 main device may continue the communication with the computer C. Communication with the computer C without being passed through the tunnel may be stopped a definite time after the power turn-off. Or if the address of the terminal device
25 contained in a session connection request received from the terminal device is different from that contained in a session connection request received last time, communication with the computer C without being passed through the tunnel may be stopped.

30 According to the embodiments of the present invention, the main device can transmit/receive broadcast packets or multicast packets to/from a communication apparatus located in the neighborhood of the terminal device as heretofore described. As a result, it becomes possible for the main device to find a
35 device or service in the neighborhood of the terminal device. Furthermore, since communication started by the main device is

conducted without being passed through the terminal device, it is possible to prevent extra processing load from being imposed on the terminal device. In addition, the main device transmits a packet that is a response to a packet received without being
5 pass through the tunnel, without passing it through the tunnel. Therefore, it becomes possible to hold down the processing load required to transfer packets to the terminal device to a low value.